4
Processamento da Linguagem Natural

O Processamento de Linguagem Natural (PLN) é o campo da ciência que abrange um conjunto de métodos formais para analisar textos e gerar frases em um idioma humano. Em mineração de textos, os métodos para analisar textos escritos são usados na etapa de preprocessamento de forma a melhor representar o texto e aproveitar mais o conteúdo. O principal objetivo do PLN para essa etapa consiste em Reconhecer e Classificar as Entidades Mencionadas. Porém, para essa tarefa ser bem feita é necessário resolver concomitantemente outras tarefas de PLN. A Figura 20 mostra como é necessário utilizar técnicas conjuntas de PLN para assim melhorar o desempenho de cada uma delas separadamente.

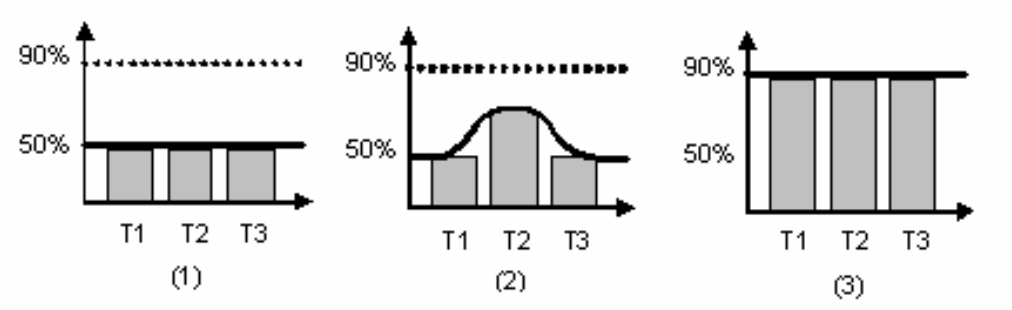

Figura 20 – Os três gráficos (1), (2) e (3) mostram, de forma ilustrativa, a necessidade colaborativa de três tarefas de PLN T1, T2 e T3. T2 só consegue atingir 90% de acerto de melhorar T1 e T3.

Em geral, a eficiência da extração das entidades de um texto se comporta de forma exponencial com a complexidade algorítmica. Com heurísticas simples atingimos facilmente um acerto de 80%. A dificuldade está em atingir valores acima de 90%, sabendo que 100% é praticamente impossível.

4.1.
O Modelo de PLN

O modelo de NLP utilizado na etapa de pré-processamento descrito nesse capítulo é fortemente baseado em léxico. O léxico utilizado no modelo contém atributos relativos a uma ontologia primitiva. Esse mesmo léxico é ainda

atualizado de forma automática. Os algoritmos de aquisição utilizados na sua atualização, assim como os algoritmos de text mining são avaliados segundo medidas de Precisão e Recordação. Nas seções seguintes veremos alguns conceitos sobre a orientação e propósito do modelo utilizado.

4.1.1. Aquisição Automática

A aquisição automática das entidades (base de conhecimento) através de PLN funciona de forma biunívoca. Para reconhecer bem os objetos de um texto é necessário uma base de conhecimento de objetos. Essa base, por sua vez, é proveniente do reconhecimento de objetos anteriores. Seguindo esse raciocínio, se temos uma regra que usa o conhecimento de objetos para reconhecê-los e extraí-los. É necessária uma outra regra mais forte (menos ambígua) que aponte de forma ditatorial a ocorrência desse objeto. Essas duas regras atuarão em trechos diferentes do corpus. Exemplo, em um momento temos a ocorrência livre em um texto "Tom Cavalcanti trabalhou na Rede Globo". Usamos o léxico (regra de força 2) para nos dizer que Tom Cavalcanti é uma pessoa e recortá-lo separadamente. Porém, em outro texto encontramos "A apresentação do comediante Tom Cavalcanti será de noite" que ativa uma regra de força 1 de aquisição lexical do lexema Tom Cavalcanti classificado como pessoa para ser utilizado na regra 2.

4.1.2. O Léxico

O sistema de PLN descrito nesse capítulo conta com um léxico para armazenar as informações lingüísticas da língua. O léxico é uma tabela onde cada registro dessa tabela apresenta um lexema e seus atributos. A idéia subjacente a esse léxico é de que cada registro apresente um lexema que possua um significado específico. Um lexema, como veremos a seguir, é uma seqüência de caracteres que pode conter qualquer caractere incluindo o espaço (" "), a exemplo "maça verde". Devem existir tantas ocorrências do mesmo lexema, quantos forem os significados possíveis. Sendo assim, um registro pretende representar um significado e a representação morfológica dele. Cada significado recebe um identificador.

Porém, existe o caso inverso, onde duas seqüências diferentes apontam para o mesmo significado. Esse problema é contornado pela adição de um campo (meaningId) agrupando os identificadores. Além desse campo, ainda são armazenados os identificadores do lexema, a seqüência de caracteres, a classe, a freqüência e idioma como exemplificados na Tabela 3.

Tabela 3 – Especificação da tabela e exemplos de registros do léxico utilizado

meaningId	lexemeId	Keylexem	class	Freq	Idiom
356	12	Plant	S	3500	E
356	13	Vegetal	S	56768	E
432	14	Plant	S	270	E
432	15	industrial plant	S	1390	E

Na transição da etapa de pré-processamento para a indexação, os keylexems gerados pelo PLN e armazenados no léxico passam a se chamar tecnicamente de índice invertido. Isso quer dizer que o sistema de indexação que será usado contém um índice estendido, com todos os atributos do léxico, diferentemente dos tradicionais com apenas keyword e freqüência.

4.1.3. Sobre a Delimitação da Unidade Lexical

Se o léxico computacional é o repositório das palavras de um sistema de PLN, nessa seção discutiremos como serão definidos seus itens (ou itens lexicais como são mais comumente chamados). Esta reflexão é necessária já que manter em uso termos cujos conceitos não são satisfatórios é se propor a chegar a resultados questionáveis. Por isso uma rápida reflexão sobre este ponto é fundamental.

Serão estabelecidos os limites e características da unidade lexical, que será nosso objeto de manipulação. O termo comumente usado para tratar desse objeto é palavra-chave (em inglês, keyword). Porém keyword trata, em última instância, de uma seqüência ininterrupta de caracteres alfabéticos e hífens. Essa definição, apesar de sua simplicidade, pode nos trazer dificuldades na composição dos significados.

Como os modelos descritos nesse capítulo são fortemente baseados em um léxico orientado por significados, vamos definir um conceito de palavra que será chamado de keylexem (em analogia a keylexeme, ao invés de keyword) que é uma seqüência de caracteres armazenada em uma registro lexical. Um keylexem pode tanto ser "bola" como "à frente do tempo", "fim de semana", "Fernando H. Cardoso" ou até "Dois Filhos de Francisco".

O conceito de keylexem será utilizado daqui em diante nas etapas de preprocessamento como referência a um objeto lingüístico formado por uma seqüência de quaisquer caracteres.

4.1.4. Ontologia

Uma ontologia é um conjunto de conceitos padronizados, termos e definições aceitos por uma comunidade particular. Ela inclui a definição desses conceitos, suas propriedades e as restrições entre os mesmos. A mais freqüente definição de ontologia é a de Gruber (Gruber, T. R., 1993) "uma ontologia é uma especificação de uma conceituação". Seu uso tem sido crescente no âmbito da Web Semântica.

De outra forma, uma ontologia pode ser entendida como um modelo lógico de representação do mundo (Guarino, N., 1998). Por ser baseada em lógica, facilita a implementação computacional. Porém, como toda representação, uma ontologia expressa apenas uma visão, um recorte, uma projeção do mundo, não pretendendo dar conta dele por inteiro.

4.1.5. Precisão e Recordação

Precisão (*precision*) e recordação (*recall*) são as medidas básicas usadas na avaliação de eficiência de sistemas tanto para busca quanto para aprendizado. A busca do tipo precisão é adequada quando se sabe exatamente o que se quer. A busca do tipo recordação é adequada quando não se sabe o que se quer e precisa-se fazer uma exploração, um reconhecimento de um domínio para então decidir o que se quer. Em alguns casos será necessário utilizar em seguida uma busca do tipo Precisão, em outros obter-se-á o item desejado diretamente.

Tecnicamente a busca Precisão tende a retornar poucos documentos e a busca Recordação a retornar muitos documentos. Para aprendizado as medidas funcionam de forma análoga, trocando os itens por lexemas. Ao invés de retornar documentos em uma busca, seleciona lexemas para aquisição. Um sistema de aprendizado orientado a precisão dificilmente comete erros, porém demora a aprender novos lexemas. Um sistema voltado a recordação aprende mais conhecimento em menos textos, porém pode cometer mais erros.

Na prática, existe um contra-peso entre precisão e recordação. Ao maximizar a precisão, perde-se recordação. Ao maximizar recordação perde-se precisão. No entanto, o principal objetivo é manter as duas medidas altas reforçando uma ou outra dependendo da funcionalidade da aplicação. Se retornamos todos os documentos maximizamos a recordação e minimizamos precisão. Se trouzermos só um documento, correto, maximizamos precisão e minimizamos recordação formando uma estreita visão da base. Se estamos fazendo os primeiros passos da investigação, devemos reforçar a recordação. Se já sabemos o que queremos podemos reforçar a precisão.

Cada uma das técnicas de PLN descritas a seguir têm impacto nessas medidas, tornando o sistema mais orientado à precisão ou recordação. Os cáculos para essas medidas são descritos de forma aprofundada em (Baeza-Yates, B. e Ribeiro Neto, B., 1999).

4.2. Técnicas de PLN

4.2.1. Tokenização

A tokenização é o primeiro estágio do pré-processamento de um texto. Nele, o texto representado por uma seqüência de caracteres é agrupado em um primeiro nível segundo fronteiras delimitadas por caracteres primitivos como espaço (“ ”), vírgula, ponto etc.

Cada grupo de caracteres estabelecidos no primeiro nível é chamado de token. A seqüência desses grupos, por sua vez, é chamada de tokenstream. Tanto os grupos de caracteres, como os delimitadores se tornam tokens na nova seqüência, o único caractere descartado é o espaço em branco.

O jogador, que está de camisa verde, marcou o gol da vitória.

[O] [jogador] [,] [que] [está] [de] [camisa] [verde] [,] [marcou] [o] [gol] [da] [vitória] [.]

O resultado desse processo na língua portuguesa é uma seqüência de palavras intercaladas por espaço e algumas vezes por símbolos delimitadores. Apenas com esse primeiro estágio já é possível iniciar um processo de indexação para recuperação de informações. Veremos que, se essa estratégia for usada, além de armazenar tokens desnecessários como "(-):,/", encontramos rapidamente problemas de excesso de precisão, pois os verbos e nominalizações devem aparecer exatamente da forma digitada, sem generalização, dificultando a busca. Esse problema inclui letras maiúsculas e minúsculas, ao buscarmos pela palavra "dicionário" não chegaremos até o conteúdo contendo "Dicionário".

Os caracteres delimitadores armazenados que se apresentam em grande quantidade e muitas vezes desnecessários para a representação do conteúdo podem ser descartados (como feito com os espaços em branco), porém, em alguns momentos eles podem assumir papéis críticos para a busca. O ponto, por exemplo, pode assumir papel de fim de sentença, URL, data, número etc. Apesar da maioria dos pontos marcarem fim de sentença, um sistema eficiente deve resolver outros casos além desse.

4.2.2. Normalização

É uma técnica para aumentar a Recordação em virtude das diversas representações de um mesmo conceito. A idéia é esquivar das várias formas de representação de uma palavra associada a um mesmo conceito. Por exemplo, do conceito de "objeto físico que consiste em um número de páginas atadas juntamente" temos a palavra "livro" com as seguintes representações "livro" e "livros" (plural). O processo de normalização propõe que essas duas formas sejam agrupadas em apenas uma, indicando que, para a busca, elas têm o mesmo significado. A questão da normalização é que ela é, justamente, uma aproximação de conceitos, ou seja, os lexemas não têm o mesmo significado e sim um alto grau de redundância de significado, que, para uma estratégia do tipo Recordação, pode ser interessante.

Na prática, vemos que ao aumentar a Recordação dessa forma agruparemos várias palavras que possuem significados distintos, prejudicando bastante a precisão do sistema. Porém, a estratégia do tipo Recordação, por reduzir o tamanho do léxico, normalmente apresenta uma maior eficiência quando o objetivo é navegação.

De acordo com a forma de agrupamento das realizações das palavras, os processos de normalização podem ser de vários tipos. Alguns deles são:

Lematização: Substitui-se as diversas formas de representação da palavra pela forma primitiva. As formas "livro", "livros" e "livraria" apontam todas para o lexema "livro". Essa estratégia tem a vantagem de captar mais as intenções de busca do usuário, já que a forma livro é de fácil interpretação.

Radicalização Inflexional (*stemming*): Só leva em consideração as flexões verbais. Faz truncamentos que tornam as palavras, na maioria das vezes de difícil compreensão. No mesmo exemplo: "livro", "livros", "livrando" é substituída por uma quarta forma "livr".

Radicalização para a raiz (*stemming*): É a forma mais agressiva de normalização, levando em consideração todas as formas de sufixos e de prefixos. A eficiência desses métodos podem ser vistas em (Kantrowitz, M. et al, 2000).

Sinônimos: No senso comum, os sinônimos são palavras bastante diferentes que contêm o mesmo significado, ou apontam para o mesm1o objeto. Se as duas ocorrências forem agrupadas por um único identificador teremos também um procedimento de normalização.

Porém, alguns lingüistas afirmam que não existem sinônimos exatos, i.e., que guardam consigo exatamente o mesmo significado. Assim, a idéia mais adequada a relação de sinonímia seria a de similaridade, ou, a de redundância de informação. Então, se os significados são parecidos quer dizer que existe uma parte diferente. Mais, essa parte diferente pode variar em cada caso, mantendo ainda a relação de sinonímia.

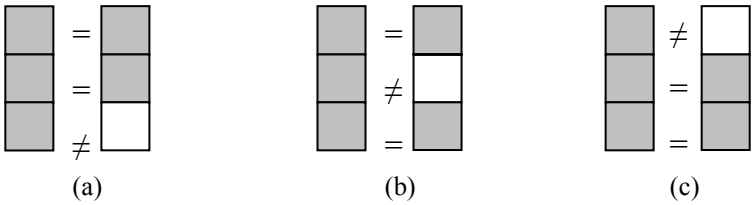

Figura 21 – As figures (a), (b) e (c) ilustram três relações diferentes e hipotéticas de sinonímia.

Para cada parte diferente indicada na Figura 21 temos um tipo de relação de sinonímia. Essas relações permitem fazer um verdadeiro passeio semântico para ilustração, onde cada palavra apresenta um interseção de significado com a outra. Vamos demonstrar isso com a palavra "planta" que pode ser uma "representação arquitetônica", um "arbusto", uma "flor", um "vegetal" ou até o verbo "plantar". Porém, nenhuma delas pode ser exatamente igual a "planta", por que senão, pela lógica implicaria que todas as palavras são iguais entre si, e definitivamente um "arbusto" não é o mesmo que uma "flor", muito menos igual a uma "fábrica".

4.2.3. Expressões Multi-Vocabulares

Geralmente, itens lexicais são palavras geradas pelo processo de tokenização. Mas, existem vários casos onde ajuda considerar um grupo de palavras como um único item lexical. Na literatura lingüística temos algumas classes desses grupos que são colocações, expressões e expressões idiomáticas.

De acordo com (Saeed, J. L., 1997), a diferença entre colocação, expressão e expressão idiomática está relacionada ao dinamismo da linguagem. Dessa forma, colocações são as combinações mais dinâmicas de palavras. Conforme elas vão sofrendo um processo de fossilização se tranformam em expressões. Expressões idiomáticas, a partir desse ponto de vista, são os tipos mais fossilizados de combinações, de forma que as palavras combinadas se "desgarram" do significado das palavras individuais da combinação. Isso quer dizer que a diferença entre esses termos está relacionada com o uso. Exemplos dessa dinâmica são "Vossa Mercê", "Você" e "cê", ou mesmo "em boa hora", "embora" e "bora". A freqüência de uso durante uma considerável quantidade de

tempo seria responsável pelo mudança no estatus de palavra para expressões multi-vocabulares. Por esse motivo, as abordagens estatísticas têm se destacado bastante na solução desses problemas.

Uma vez que uma expressão multi-vocabular é detectada, ela é inserida no léxico e se torna um keylexem. As expressões multi-vocabulares fazem retomar a importância do espaço em branco. Além disso, por analisar um conjunto de palavras próximas em significado, esse procedimento pode ser encarado como um passo em direção a desambigüização, já que seu significado é distanciado quando da ocorrência da palavra agregada.

4.2.4. Fronteiras das Frases

Apesar de intuitivamente parecer simples considerarmos tudo que está entre os delimitadores (.!?), essa abordagem não mostrou-se muito boa, com uma primeira aproximação do problema de apenas 80%. Para superar a barreira dos 95% devemos desambiguizar os delimitadores, isto é, classificá-los como delimitadores ou não-delimitadores. Esse é um dos procedimentos mais difíceis, e, de certa forma são resolvidos pouco a pouco durante as outras etapas do pipeline de PLN. Assim, uma melhor aproximação para a determinação das sentenças é deixar essa etapa para o final, depois de ter resolvido as outras. A maioria dos delimitadores, como o caso do ponto, já vai estar agrupada em nomes, urls e datas. O pontos restantes promoverão um acerto muito maior.

Além do problema da ambigüidade dos delimitadores, existem ainda outros desafios na definição das fronteiras das frases (em inglês, Sentence Boundary). Primeiramente o problema da ocorrência dos operadores “” e (). Existem frases que contêm frases dentro delas de forma recursiva, e, nestes casos, o ponto não mais promove o fim da frase.

“Eu não vou com você. Me liga quando chegar lá.” Disse Márcio Lemos.

Teremos que ter estruturas em árvore para representar a delimitação das sentenças de um texto. Adicionalmente temos ainda as estruturas de parágrafo, capítulo e documento. (Corpus Encoding Standard Annex 10)

4.2.5. Etiquetagem

A etiquetagem é a rotulação das palavras de acordo com uma dada classificação que pode atender à uma ontologia. No nosso caso uma estrutura de representação básica presente em quase todas as línguas, uma ontologia considerada neste trabalho como primitiva. Essa ontologia primitiva começa com a divisão em duas categorias principais:

- **Palavras funcionais**: a categoria engloba as preposições, artigos, pronomes, etc. Esse conjunto é estático, pois raramente surge uma nova palavra funcional em uma língua, por contrapartida essas palavras aparecem mais freqüentemente em um texto. É raro encontrarmos um texto que não tenha a ocorrência do lexema "de".
- **Palavras de conteúdo**: Diz respeito aos nomes, verbos, adjetivos etc. Esse conjunto é dinâmico, pois a todo momento surge uma nova palavra de conteúdo em uma língua, por contrapartida essas palavras aparecem menos freqüentemente em um texto. É raro encontrarmos em um corpus genérico vários textos com a palavra "ontologia".

As palavras funcionais, por serem mais estáticas, podem ser catalogadas manualmente em um léxico. Já as palavras de conteúdo devem ser aprendidas/inseridas constantemente e, por isso, de preferência de forma automática. Isso não significa descartar um dicionário de palavras de conteúdo. Os dicionários são úteis, porém não são suficientes. Por exemplo, a palavra "livro" pode ser um substantivo ou um verbo. A tarefa da etiquetagem é justamente classificá-la corretamente.

Para as palavras de conteúdo precisamos, então, de uma abordagem de aquisição automática das entradas do léxico, assim como sua respectiva classe. Na literatura tem-se usado algoritmos de aprendizado de máquina como HMM (Hidden Markov Models – Rabiner, 1989) e TBL (Transformation Based Learning – Brill, 1992) que aprendem através de um corpora etiquetado.

Uma boa abordagem para adquirir a classificação de palavras de conteúdo é começar pela distinção majoritária entre substantivo e verbo. Os substantivos, em geral, nomeiam os objetos enquanto os verbos especificam as ações e relacionamentos. Para algumas línguas latinas esse problema é mais fácil pois os verbos apresentam sufixos sofisticados que indicam a classificação das palavras na maioria dos casos. Já para a língua inglesa a tarefa não é tão simples, a palavra substantivo ou verbo costuma aparecer na mesma forma o que força a antecipação da tarefa de desambigüização do sentido das palavras.

Estratégias baseadas em regras são adequadas se for considerada a classe de palavras vizinhas e costumam obter um bom índice de acerto. Para essa estratégia é necessário usar um dicionário para dar uma condição inicial de classes ao sistema e assim gerar insumos para aplicar as regras. Na primeira aplicação as regras alteram algumas classes. A nova seqüência de classes é usada como prancheta para um nova aplicação das regras, e assim por diante. Um exemplo desse método pode ser encontrado em (Brill, E., 1995).

4.2.6. Padrões Gramaticais

A técnica de padrões gramaticais pode ser entendida como reconhecimento de padrões em um caso mais geral ou como um procedimento de parsing mais específico (ad-hoc). A técnica visa procurar no texto padrões sintáticos fortes que assumem significados próprios e, por isso, merecem ser uma unidade de lexema como por exemplo “através de”, “de acordo com” e “faz parte” que seriam proveniente de seqüências de classes gramaticais. Em casos mais genéricos essas sequenciaw podem incluir delimitadores, assim temos a presença de padrões como “’s”, “--“ ou até “:-)”.

4.2.7. Reconhecimento de Entidades Mencionadas

O pré-processamento comum à maioria das atividades em mineração de textos tem por responsabilidade principal o reconhecimento das entidades mencionadas no texto. Por entidades podemos entender pessoas, lugares, instituições etc. Porém, veremos que para reconhecer essas entidades de forma eficiente faz-se necessário o reconhecimento de todos os objetos do texto.

A teoria em torno do processo de identificação e classificação de entidades tem como referência (Borguraev, B. e Pustejovsky, J., 1996), onde eles descrevem, para a língua inglesa, o processo de segmentação de nomes próprios apontando algumas dificuldades. Um procedimento natural para humanos mostrou-se uma difícil tarefa para um sistema especialista. Destacam-se alguns exemplos:

(1) Philip B. Morris
(2) Juiz Nicolau dos Santos Neto
(3) Presidente da Câmara dos Vereadores Alcides Barroso
(4) Hollywood

No caso (1) temos um nome próprio não trivial porque contém uma abreviação no meio que poderia ser considerado como ponto final. No caso (2) temos um item funcional "dos" que poderia separar o nome em dois distintos: "Juiz Nicolau" e "Santos Neto". O caso (3) contém de fato duas entidades, sendo que não há nenhuma evidência de onde segmentar. Além disso, para classificação teremos o problema da polissemia, onde a entidade caso (4) "Hollywood" pode significar o lugar ou a marca de cigarros.

Estes exemplos servem apenas para destrivializar o pré-processamento e mostrar que nesse caso os procedimentos vão muito contra nossa intuição, onde reconhecer EM não é apenas um processo de recortar o que tem letra maiúscula.

A seguir descreveremos o módulo de inferência que identifica e classifica as entidades (para o idioma português brasileiro) utilizado nessa tese. É de grande eficiência e se baseia em conhecimento lingüístico, métodos estatísticos e modelos cibernéticos.

O reconhecimento de entidades mencionadas (em inglês, Named Entity Recognition) é um dos pontos principais do PLN para inteligência competitiva pois eles nomeiam os objetos do mundo real de trabalho. Além disso, grande parte da informação de uma nova notícia é proveniente de novos nomes, ou relacionamentos entre novas combinações de nomes. Os tipos de relacionamentos são mais finitos que os nomes, aproximadamente 90% dos novos lexemas a serem aprendidos por um sistema automático são nomes próprios. Sendo assim, é interessante dar especial atenção a tarefa de reconhecimento de entidades.

O processo começa pela avaliação dos candidatos a entidade nomeada. De forma macro, essa avaliação consiste em uma sucessão de filtros. Os candidatos que persistirem serão agrupados por proximidade e considerados nomes de entidades. Esses filtros algumas vezes utilizam o condicionamento a palavras próximas, comportando-se como um autômato finito.

Um bom marcador utilizado é a letra maiúscula. Ela costuma fornecer uma boa lista de candidatos iniciais. Mas esse é só o começo da solução. Além disso, um algoritmo específico para avaliar o início de uma frase deve ser utilizado, já que todas as palavras, inclusive os nomes próprios, são marcadas com letra maiúscula no inicio das frases. Uma boa solução para o início de frases é saber se a palavra é um verbo ou um substantivo antes de tornar o token candidato a ser um nome próprio. Deve-se considerar também o fato de que um token nome próprio é encontrado bastante freqüentemente acompanhado por outro token nome próprio.

"Sônia Braga", "São Pedro da Aldeia", "Manhattan Connection"

Esses tokens em seqüência normalmente representam um único lexema e devem ser agrupados. Deve-se levar em conta as preposições também, mesmo que não sejam marcadas pela letra maiúscula. Outro problema freqüente é a existência de tokens periféricos responsáveis pela qualificação da entidade e que também são marcados pela letra maiúscula.

"Presidente Lula", "Estado do Rio de Janeiro"

Outro padrão recorrente é o uso de siglas no meio do nome como forma de abreviação. Os pontos utilizados nessas abreviações são um grande complicador para o reconhecimento. Saber se "Murilo O. Machado" é um nome ou "Murilo O" é o fim da frase e "Machado" o começo de outra pode parecer trivial para nós humanos mas um computador deve ter regras que auxiliem nesse agrupamento como a raridade de uma frase terminando em sigla, e ainda mais precedido de um nome de pessoa.

Finalmente, passamos esses objetos por um último filtro de datas, religião e localização geográfica e rotulamos como nomes de entidades. Esses nomes

devem ser catalogados e aprendidos periodicamente para auxiliar as outras tarefas de PLN.

4.2.8. Classificação de Entidades Mencionadas

A classificação de entidades mencionadas (em inglês, Named Entity Classification) é realizada nas entidades reconhecidas (portanto do resultados do procedimento de reconhecimento de entidades mencionadas). O objetivo é classificar os nomes, que são os principais objetos da IC, segundo um ontologia.

A Figura 22 mostra as classes utilizadas nos processos de reconhecimento e classificação das entidades mencionadas, assim como o apoio fornecido pelo conhecimento de uma na outra. Exemplo, as regras de classificação de datas usam perguntas sobre a classificação dos números. As classes que se encontram no nível mais baixo da árvore tendem a ser mais independente da língua, como os nomes de pessoa e de organização, em seguida temos os nomes de locais geográficos e finalmente os formatos de datas.

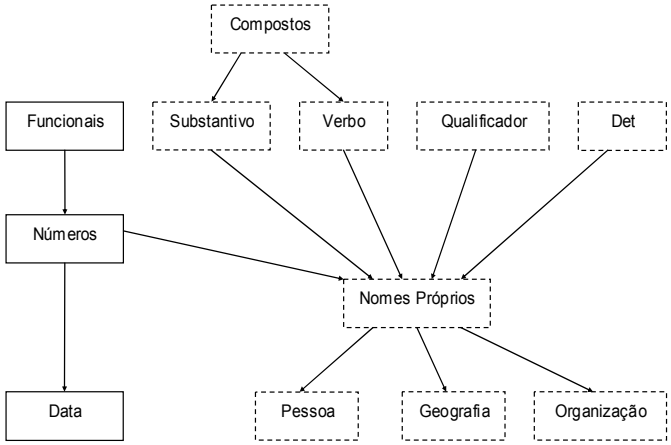

Figura 22 – Esquema de pré-requisitos entre as classes ontológicas.

O processo de classificação pode ser feito seguindo a mesma lógica do processo de reconhecimento, porém em um nível de abstração maior. A maioria das regras contém perguntas sobre o resultado do reconhecimento. Além disso, existe a meta informação. São utilizadas também regras que utilizam de palavras adjacentes que circunscrevem a classificação dos nomes.

"Isso aconteceu próximo litoral da ilha de Sumatra"

Nesse exemplo, as palavras "litoral" e "ilha" ditam a classificação de localização geográfica para o nome "Sumatra".

4.2.9. Análise dos Constituintes

Até então, as técnicas vistas descrevem processamentos que consideram apenas as palavras individualmente ou expressões que funcionam como itens lexicais, de forma a enriquecê-las e transformá-las em um lexema dotado de significado próprio. A análise dos constituintes (*parsing*) atua sob o resultado gerado pelas etapas anteriores com o objetivo de ligar as palavras umas as outras, estabelecer relações entre elas. Essas relações são especificadas pelo próprio texto. Se o texto estiver escrito corretamente e tratar do mesmo assunto, nenhuma palavra ficará solta, e todas elas terão uma posição específica em um único grafo. Mais precisamente, cada palavra tem um papel dentro da frase que se encontra; cada frase pode ser representada em uma estrutura sintática de árvore; as árvores são conectadas entre si por meio de relações de anáfora, hiperonímia, sinonímia etc formando o grafo final. Um texto que resulta em dois grafos não conectados contam duas histórias diferentes.

Em (Grune, D. e Jacobs, C. J. H., 1991) são apresentados vários tipos de *parsing* assim como algumas abordagens para o problema. Um dos métodos computacionais para se aproximar desse problema é a estrutura de DCG (Defined Clause Grammar) que combina uma seqüência de regras de formação das árvores a partir dos constituintes.

S → SN SV
SN → DET N
SV → V SN

Essas regras operam com classes atribuídas aos lexemas nas etapas anteriores. São armazenadas em um banco de inferência. Um dos problemas dessa abordagem é que o número de regras pode crescer de forma indefinida, tentando atender ao número de formas sintáticas possíveis em uma língua, que podem ser

infinitas. Além disso, é necessário manter um motor de inferência verificando a integridade das regras a cada vez que uma regra é adicionada, pois essa regra pode entrar em contradição com outra ou uma combinação lógica de outras.

Diante desses problemas, o procedimento de análise dos constituintes só é utilizado para casos específicos e domínios com alguma restrição. Mesmo com essa limitação, o procedimento atende bem a funcionalidade de information extraction, pois possui os objetos e relacionamentos pré-especificados, restando ao analisador detectar as variações sintáticas.

Sob o ponto de vista da aquisição lexical, podemos levar em conta a freqüência de uso de cada regra como estimador de sua força de atuação. Se uma regra do tipo DET + S ocorre muito na língua, então, se encontrarmos um DET e a palavra seguinte tiver sido avaliada como V (verbo) pode-se re-rotular a classe dela para S. Deve-se tomar bastante cuidado com a utilização de regras de re-rotulamento pois elas aumentam de forma significativa a complexidade computacional do problema.

4.2.10. Correferência

Assumindo a representação de um texto como um grafo estruturado por árvores interconcetadas, onde cada árvore representa um relacionamento parseado. A maioria das correferências são formas de conexão entre as árvores. Elas ocorrem quando dois termos de um texto têm o mesmo referente, i.e., se referem ao mesmo objeto/entidade. Elas também ocorrem freqüentemente usando a relação ontológica hierarquicamente superior. As correferências são usadas quando a definição de um objeto tem uma relação de dependência conceitual de um objeto já instanciado. A seguir encontram-se abordagens de solução de alguns tipos mais freqüentes de correferência.

4.2.10.1. Acrônimos, Siglas e Abreviaturas

Os acrônimos são palavras formadas por pedaços de nomes extensos como “Pontifícia Universidade Católica” e PUC. O dois keylexems são distintos e se referem exatamente à mesma entidade que é uma universidade com um

determinado número de habitantes e uma certa cultura. Em um texto, o acrônimo costuma aparecer marcado por parênteses.

Padrão 1: Departamento de Engenharia Elétrica (DEE) ou
Padrão 2: DEE (Departamento de Engenharia Elétrica)

Essa é uma forma de aprender automaticamente a conexão entre os keylexems, no entanto, existem vários casos em que os parentes aparecem e não há acrônimos. Temos que investigar outras dicas que possam fortalecer a regra. Primeiramente descobrimos se o possível acrônimo se encontra no padrão 1 ou no padrão 2. A idéia é usar o fato de o nome ser sempre maior que o acrônimo, e, com o candidato acrônimo em mãos, tentamos casar (match) as letras com as iniciais do nome. Essa comparação deve gerar um score de probabilidade de existir a conexão.

As siglas diferenciam-se dos acrônimos pela intercalação de pontos (ex. E.U.A.) e normalmente aparecem dentro dos nomes, como "George W. Bush". Quando não, podem ser comparadas com as entidades presentes no textos ou muito freqüentes.

Finalmente, as abreviaturas são nomes truncados por pontos. A maioria é muito freqüente e pode ser catalogada, como "Dr.", "Ltda." e "Av.", já as mais raras devem ser comparadas com palavras longas e bastante usadas para sugerir um aprendizado automático.

4.2.10.2. Nomes Truncados

Os nomes truncados são as correferências mais freqüentes, principalmente do domínio da inteligência competitiva. A ocorrência desse fenômeno é proveniente de um princípio da economia na comunicação. Se a entidade com o nome inteiro já foi apresentada, não é mais necessário repetir toda a informação. No caso dos nomes de pessoa, uma vez apresentado o nome "Sônia Braga", será referenciado mais a frente por Sônia ou Braga e não pelo nome inteiro. Essa ligação tem ainda uma outra característica adicional, ela é valida só no contexto (localmennte). O único keylexem gerado é "Sônia Braga" ligado aos nomes truncados que serão normalizados pelo nome inteiro.

Pelo valor da informação presente nos nomes de pessoa e pela freqüência em que isso ocorre é recomendado uma atenção especial para esse processamento. Depois de reconhecidas e classificadas todos as entidades dos textos, deve-se separar apenas os nomes de pessoas e fazer uma comparação direta de todas as combinações de duplas. Os nomes que se encaixarem perfeitamente (excluindo a técnica de identificação de erros ortográficos para simplificar) são candidatos a correferência. Em um segundo momento deve-se levar em conta a distância dentre os dois nomes em número de palavras, se a distância for muito grande pode ser uma correlação espúria devido ao fato de que nós mesmos teríamos muita dificuldade em lembrar do nome citado. A distância também pode servir para resolver um problema de dupla referência, quando duas pessoas tem o mesmo nome ou sobrenome. Finalmente, a seqüência de apresentação do nome inteiro e depois a truncado é muito mais usual, devendo ganhar um peso maior, apesar de o contrário também ocorrer.

4.2.10.3. Anáfora Pronominal

As anáforas são as correferências formadas por palavras funcionais, mais freqüentemente por pronomes. Essas correferencias servem para adicionar informação descritiva ao objeto lingüístico. Como não guardam nenhum conteúdo ortográfico da entidade em sua representação, são usadas com bastante proximidade ao objeto referenciado. Porém, mesmo próximas, ainda é difícil o trabalho de encontrar o objeto destino correto. Essa decisão deve ser feita na comparação de atributos lingüísticos do objeto, por exemplo, “Ela” deve referenciar um objeto pessoa feminino singular; “Eles” deve referenciar um objeto pessoa plural e assim por diante.

A vantagem é que os pronomes são finitos e catalogáveis, sendo viável construir uma matriz de atributos de comparação para cada um. Mais complicado é incorporar essas características na etapa de classificação das entidades. Alguns trabalhos tentam resolver esse problema, a exemplo (Lappin, S. e Leass, H., 1994) e (Ge, N. et al, 1998)

4.2.10.4. Sinônimos

Os sinônimos também são muito aplicados para correferenciar termos. Nessas ocasiões, as duas palavras sinônimas, apesar de não conterem, em sua essência, significados idênticos, no contexto, apontam exatamente para o mesmo objeto manifestando uma relação de correferencia.

As relações de sinonímia dessas palavras, em geral, são especiais e apresentam na redundância de informação um certo grau de hierarquia de organização do conhecimento como "tipo de" ou "parte de". Apesar de que sinônimos horizontais também são encontrados.

Em geral os textos apresentam esse tipo de correferência para não repetir a mesma palavra. Exemplo, um texto pode começar falando de uma pesquisa, e depois correferenciar a mesma pesquisa como trabalho, paper ou desenvolvimento. No domínio de negócios um exemplo comum é apresentar a empresa e depois chamar de agência, corporação etc. As relações hierarquicas podem ser adquiridas de forma automática através de regras atuando em padrões como "gasolina e outros combustíveis" indicando que gasolina é um tipo de combustível. Outros padrões lingüísticos podem ser encontrados em (Hearst, M. A., 1992). Após a extração do conhecimento, a idéia é montar um banco de dados contendo um grafo dessas relações para auxiliar na resolução da correferência de novos textos.

4.2.10.5. Erros Ortográficos

Correção ortográfica automática tem uma longa história. Um importante algoritmo criado em 1964 introduziu a idéia de distância edição mínima (minimum edit). Basicamente, o conceito de distância de edição quantifica a idéia de uma seqüência de caracteres estar próxima a outra, pela contagem do número de operações de caracteres (como inserção, deleção e substituição) que são feitas para tranformar uma string em outra. Usando essa métrica, os melhores candidatos para a palavra correta são aquelas que apresentação a mínima distância de edição.

Outra abordagem é a técnica de chave de similaridade, onde as palavras são tranformadas em um tipo de chave de modo que palavras parecidas e erradas

tenham a mesma chave. Para corrigir os erros ortográficos é necessário então simplesmente gerar a chave para a palavra errada e procurar as palavras com a mesma chave para uma lista de candidatos. Soundex é a melhor solução conhecida com essa abordagem, e também é usado a aplicaçãod e busca fonética.

A combinação da distância de mínima de edição e chaves de similaridade (metaphone) é o núcleo da estratégia utilizada pela Aspell. Mas existe uma terceira abordagem utilizando técnicas de indexação por n-gramas de letras.

Uma n-grama de letras é uma seqüência de n letras da um dada palavra. Para exemplicar, a palavra "cavalo" pode ser dividida em quatro 3-gramas, também conhecido como trigramas: "cav", "ava", "val" e "alo". A idéia é que os erros ortográficos mais comuns só afetam poucos constituentes de n-grama, então, podemos buscar pela palavra correta através daqueles que compartilham a maior parte dos n-gramas com a palavra errada.

No processo de indexação normal, os documentos são indexados por palavras contidas neles. No processo de correção de erros, os documentos são as palavras e as palavras são os n-grams que irão constituir o índice. Quando a palavra é procurada, os n-grams são processados e procurados no índice, a palavra que apresentar o maior número de n-grams será a mais relevante.

O problema de misspelling é bastante útil em PLN e já apresenta bons resultados na literatura, por esse motivo vale a pena ver ainda outras abordagens como (Cucerzan, S. e Brill, E., 2004) e (Martins, B. e Silva, M. J., 2004).

4.2.11. Discriminação do Sentido da Palavra

O problema de discriminação do significado (em inglês, Word Sense Discrimination) é um dos mais requintados do PLN devido principalmente à sua dificuldade algorítmica. Apesar de ter havido um trabalho substancial nesse problema durante um bom período, ainda não existem algoritmos que possam completamente discriminar o sentido de uma palavra. No início dessa tese, foi estudado o tema para avaliar a dificuldade algoritmica, o resultado desse estudo originou um artigo publicado em TIL2004 (Aranha, C. et al, 2004).

Esse trabalho direcionou os estudos para o pré-processamento da linguagem, já que enfrentamos uma dificuldade de avaliação do resultado final

devido à dificuldade de interpretação da saída do modelo. Um pré-processamento mais apurado poderia aumentar a semântica dos resultados.

O processo de discriminação consiste em determinar qual o significado da palavra de acordo com o contexto em que ela está inserida. Esse problema existe devido ao fenômeno da polissemia presente nas línguas, i.e., uma mesma seqüência de caracteres pode apontar para vários significados diferentes. Sob um certo ponto de vista, todas as palavras são polissêmicas, e dessa forma, para processar cada palavra de um texto seria necessário um procedimento de discriminação. Porém, devido à freqüência de utilização, apesar de muitos significados, as palavras normalmente trazem consigo um significado mais forte (mais freqüente). Por esse motivo é que PLN sem procedimentos sofisticados de Word Sense Discrimination pode gerar resultados positivos em certas instâncias.

Uma versão maior do problema de Discriminação do sentido (*Word Sense Discrimination*) é o problema de Desambigüização do Sentido (*Word Sense Desambiguation*). A maioria dos algoritmos de desambiguização rotulam o nome do grupo além de discriminá-lo. Porém, algoritmos de discriminação dos significados das palavras têm a vantagem de ser mais orientados a aquisição automática por seguirem os modelos cognitivos de aprendizado, que, por sua vez, não têm acesso aos rótulos dos significados na maioria das vezes.

Uma abordagem bastante conhecida está descrita em (Manning, C. e Schutze, H., 1999), onde é elaborada uma definição de vetores de contexto baseados em coocorrência. Esses vetores são plotados em um espaço n-dimensional e agrupados por um algoritmo de clustering. Cada grupo indica um contexto discriminado.

4.2.11.1. Detecção Automática de Sinônimos

Uma abordagem baseada em regras (*rule-based*) desse problema já foi citada na seção 4.2.10.4. Nessa seção falaremos de uma abordagem estatística para tentar solucionar o problemas dos sinônimos horizontais (sem hierarquia). O problema da detecção automática de sinônimos é oposto/dual à Discriminação de Sentido. De forma didática, a discriminação de sentidos parte de uma mesma palavra com dois significados distintos e os sinônimos são duas palavras distintas

com o mesmo significado. No primeiro caso, temos o movimento de separação e no segundo, um movimento de agrupamento.

Tecnicamente, para cada ocorrência é computado um vetor de contexto. Após o algoritmo de clustering são detectados grupos de contextos semelhantes. Na tarefa de discriminação, os dois usos caem em grupos diferentes, apresentam significados diferentes. Logo, se duas palavras diferentes geram vetores de contextos do mesmo grupo (muito próximos) então pode-se aferir relação de sinonímia. A medida de proximidade estatística utilizada no algoritmo vai indicar o quão parecido são os significados. Normalmente estipula-se um fator limite (threshold) para o grau mínimos de semelhança que será considerado como sinônimo.